

REMARKS

Applicants respectfully request reconsideration of this application as amended. Claims 1, 9, 12,14, and 32 have been amended; claim 8 has been cancelled. Therefore, claims 1-7, 9-12, 14-15 and 31-35 are now presented for reconsideration and Applicants respectfully request the allowance of all claims.

35 U.S.C. § 101 Rejections

Claims 32-35 are rejected under 35 U.S.C. § 101 because the Office Action alleges that the claimed invention is directed to non-statutory subject matter. While Applicants do not agree that the claim as previously drafted was directed to non-statutory subject matter, Applicants have amended claim 32 in accordance with recent USPTO guidance on “computer readable media” claims. Specifically, the words “non-transitory” have been added to claim 32 as suggested in the memo titled “Subject Matter Eligibility of Computer Readable Media” signed by Director Kappos on January 26, 2010. Applicants believe the rejection of claims 32-35 under 35 U.S.C. § 101 has been overcome and respectfully request the withdrawal of said rejection.

35 U.S.C. § 103 Rejections

Claims 1-8 and 31-35 are rejected under 35 U.S.C. § 103(a), as being allegedly unpatentable over George Coulouris, et al., *Distributed Systems Concepts and Design, Second Ed.*, pp. 165-194 and 300-308 (Addison-Wesley 1994) (hereinafter “Coulouris”), in view of Abraham Silberschatz, et al., *Operating System Concepts, Fifth Ed.*, pp. 94, 264, 267, 270,-272, 289, 293, 402-405 and 444-445 (hereinafter “Silberschatz”) in view of Summers, U.S. Patent No. 6,098,133 (hereinafter “Summers”), and further in view of Schneier, U.S. Patent No. 5,978,475 (hereinafter “Schneier”).

Claim 1, as amended, recites:

A method comprising:
maintaining a first page table map for use in an isolated execution mode
and a second page table map for use in a normal execution mode;

restricting access to an isolated area of memory to bus cycles performed in the isolated execution mode by a processor operating in the isolation execution mode, the isolated area of memory having an associated audit log to contain hash values representing information that has been successfully loaded into the isolated area of memory, the audit log to further act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode;

dynamically swapping between the first page table map and the second page table map responsive to a change in execution mode;

identifying if an event is one of a class of events to be handled in the isolated execution mode;

asserting a selection signal to select the first page table map if the event is identified as one of the class of events to be handled in the isolated execution mode;

handling the event using a table map selected by the selection signal; determining if a current mode is the isolated execution mode; loading a set of control registers with values corresponding to the first page table map if the current mode is not the isolated execution mode and the event is one of the class; and

dispatching an exception vector after the loading is complete. (emphasis added)

Coulouris, Silberschatz, Summers, and Schneier neither individually nor when combined teach or reasonably suggest at least these limitations of claim 1. For example, the combination does not describe "the isolated area of memory having an associated audit log to contain hash values representing information that has been successfully loaded into the isolated area of memory, the audit log to further act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode" as recited by claim 1. The Office Action cites *Schneier* as describe this limitation. However, *Schneier* describes "a method and apparatus for generating a secure audit log using an untrusted machine communication with a trusted machine over a limited communications channel." (*Schneier*, 3:6-10) It further describes:

Each entry in the audit log contains the one-way hash of the previous entry. This enables an auditor to verify that every entry was written into the log after the previous entry and before the subsequent entry. Any attempt to delete entries, add entries, or modify entries in the middle of the log will be immediately noticed because the one-way hash function values will no longer be valid. (*Schneier*, 3:11-18)

Therefore, the fingerprinting as described in *Schneier* does not describe “audit log to contain hash values representing information that has been successfully loaded into the isolated area of memory.” Furthermore, Section 3 (“Method of Operation”) describes how the audit log system operates in *Schneier*, including startup, writing, posting, and closing. (*Schneier* 9:31–13:2) The audit log of *Schneier* is designed such that logging occurs between “[an] untrusted machine, U, and its logging partner (here the trusted machine T).” (*Schneier* 9:34–35) Therefore, the audit log of *Schneier* does not “prove current status of the isolated execution mode.” Thus, the combination of *Coulouris*, *Silberschatz*, *Summers*, and *Schneier* does not this limitation.

Although Applicants believe the previously amended claim 1 was allowable on the above mentioned basis, claim 1 has been amended to included the limitations previously found in claim 8 to further prosecution. The Office Action cited *Coulouris* as the basis for rejection of now cancelled claim 8. Nothing in *Coulouris* discusses or even contemplates “determining if a current mode is the isolated execution mode; loading a set of control registers with values corresponding to the first page table map if the current mode is not the isolated execution mode and the event is one of the class; and dispatching an exception vector after the loading is complete.” The cited section, “6.4 Naming and protection,” is discussing how a “service provides an identifier for each of its resources.” (*Coulouris*, pg. 178) These services allow “[c]lients [to] access resources by making requests to the service that manages them, supplying the appropriate identifies.” (*Id.*) In the sub-section on protection, *Coulouris* describes that “[t]he server authenticates the client and checks the access control list at each request.” (*Id.* at 182) This is not the same as “determining if a current mode is the isolated execution mode; loading a set of control registers with values corresponding to the first page table map if the current mode is not the isolated execution mode and the event is one of the class; and dispatching an exception vector after the loading is complete” required in Applicants’ amended claim 1. Additionally, *Silberschatz*, *Summers*, and *Schneier* do not fill this gap alone or in combination with *Coulouris*.

The combination of *Coulouris*, *Silberschatz*, *Summers*, and *Schneier* does not describe “the isolated area of memory having an audit log to contain hash values representing information that has been successfully loaded into the isolated area of, the audit log to further

act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode.” Furthermore, the combination does not describe “determining if a current mode is the isolated execution mode; loading a set of control registers with values corresponding to the first page table map if the current mode is not the isolated execution mode and the event is one of the class; and dispatching an exception vector after the loading is complete.”

For at least these reasons, the combination does not describe what Applicants’ claim 1 requires. Accordingly, Applicants respectfully request the withdrawal of the rejection of claim 1. Claims 2-7, and 31 are dependent, either directly or indirectly, on claim 1 and are allowable for at least the same reasons as claim 1.

Claim 32, as amended, recites:

A non-transitory processor readable medium comprising instructions that when executed, cause a machine to:

maintain a first page table map for use in an isolated execution mode and a second page table map for use in a normal execution mode;

restrict access to an isolated area of memory to bus cycles performed in the isolated execution mode by a processor operating in the isolation execution mode, the isolated area of memory having an associated audit log to contain hash values representing information that has been successfully loaded into the isolated area of memory, the audit log to further act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode;

dynamically swap between the first page table map and the second page table map responsive to a change in execution mode;

identify if an event is one of a class of events to be handled in the isolated execution mode;

assert a selection signal to select the first page table map if the event is identified as one of the class of events to be handled in the isolated execution mode;

handle the event using a table map selected by the selection signal;
determine if a current mode is the isolated execution mode;

load a set of control registers with values corresponding to the first page table map if the current mode is not the isolated execution mode and the event is one of the class; and

dispatch an exception vector after the load is complete. (emphasis added)

Claim 32 has been amended to include limitations similar to the limitations of claim 1.

Therefore, the same arguments as applied to claim 1 also apply to claim 32. The combination of *Coulouris, Silberschatz, Summers, and Schneier* does not describe “the isolated area of memory having an audit log to contain hash values representing information that has been successfully loaded into the isolated area of, the audit log to further act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode.” Furthermore, the combination does not describe “a non-transitory processor readable medium comprising instructions that when executed, cause a machine to” “determine if a current mode is the isolated execution mode; load a set of control registers with values corresponding to the first page table map if the current mode is not the isolated execution mode and the event is one of the class; and dispatch an exception vector after the load is complete.” For at least these reasons, the combination does not describe what Applicants’ claim 32 requires.

Accordingly, Applicants respectfully request the withdrawal of the rejection of claim 32. Claims 33-35 are dependent on claim 32 and are allowable for at least the same reasons as claim 32.

Claims 9-12 and 14-15 are rejected under 35 U.S.C. §103(a), as being allegedly unpatentable over *Takahashi*, U.S. Patent No. 5,615,263 (hereinafter “*Takahashi*”), in view of *Silberschatz*, in view of *Summers*, and further in view of *Schneier*.

Claims 9-12 and 14-15 are alternatively rejected under 35 U.S.C. §103(a), as being allegedly unpatentable over *Poisner*, U.S. Patent No. 5,729,760 (hereinafter “*Poisner*”), in view of *Silberschatz*, in view of *Summers*, and further in view of *Schneier*.

Claim 9, as amended, recites:

An apparatus comprising:

 a first storage location storing control data for a first page table map for use in an isolation execution mode;

 a second storage location storing control data for a second page table map for use in a normal execution mode;

 a selection unit to select which page table map is applied responsive to receipt of an event, the selection unit to dynamically swap between the first page table map and the second page table map responsive to a change in execution mode; and

an isolated execution circuit at a processor to generate isolated access bus cycles to permit the processor to access an isolated area of memory and operate in the isolated execution mode, and further to restrict access to the isolated area, the isolated area of memory having an associated audit log to contain hash values representing information that has been successfully loaded into the isolated area of memory, the audit log to further act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode,

wherein isolated access bus cycles are to be used if the apparatus operates in an isolated execution mode. (emphasis added)

Claim 12, as amended, recites:

A computer system comprising:

a processor executing in one of a normal execution mode and an isolated execution mode associated with an isolated area of memory;

a first set of control registers to define a current memory map of the platform;

a mapping unit to dynamically load the first set of control registers responsive to an event if the event should be handled using an alternate memory map, the mapping unit including a second set of registers having a first subset corresponding to control register values for a normal execution mode memory map and a second subset corresponding to control register values for an isolated execution mode memory map, the mapping unit further including a selection unit to select and dynamically swap between the first subset and the second subset, the isolated area of memory having an associated audit log to contain hash values representing information that has been successfully loaded into the isolated area of memory, the audit log to further act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode; and

an isolated execution circuit to generate isolated access bus cycles if the processor is executing in the isolated execution mode, the isolated execution circuit to permit the processor to access the isolated area to operate in the isolated execution mode, and further to restrict access to the isolated area.

Claims 9 and 12 contain limitations similar to claim 1 above, and the arguments applied to claim 1 therefore apply to claims 9 and 12. The combinations applied to claims 9-12 and 14-15 do not describe “the isolated area of memory having an audit log to contain hash values representing information that has been successfully loaded into the isolated area of memory, the audit log to further act as a fingerprint that identifies the information loaded into the isolated area of memory, the audit log to further prove current status of the isolated execution mode.”

As such, claims 9 and 12 are allowable for at least the same reasons stated with regard to claim 1. Applicants respectfully request the withdrawal of the rejections of claims 9 and 12. Claims 10 and 11 are dependent on claim 9 and are allowable for at least the same reasons that claim 9 is allowable. Claims 14 and 15 are dependent on claim 12 and are allowable for at least the same reasons that claim 12 is allowable.

Conclusion

In light of the foregoing, reconsideration and allowance of the claims is hereby earnestly requested.

Invitation for a Telephone Interview

The Examiner is requested to call the undersigned at (408) 720-8300 if there remains any issue with allowance of the case.

Charge our Deposit Account

Please charge any shortage to our Deposit Account No. 02-2666.

Respectfully submitted,
BLAKELY, SOKOLOFF, TAYLOR & ZAFMAN LLP

Date: 3/9/10

/David F. Nicholson/
David F. Nicholson
Reg. No. 62,888

1279 Oakmead Parkway
Sunnyvale, California 94085-4040
(408) 720-8300